Variational Autoencoders on MNIST

Latent Space Structure, Sampling, and Ablation Study

Mariami Shonia
Department of Mathematics
Kutaisi International University

January 30, 2026

Table of Content

Motivation

- Many real-world datasets (e.g., images) lie on low-dimensional manifolds embedded in high-dimensional spaces.
- Learning meaningful latent representations enables:
  - Data compression
  - Generative sampling
  - Smooth interpolation between data points
- Classical autoencoders learn deterministic latent codes but:
  - Do not impose structure on the latent space
  - Do not support principled data generation
- Variational Autoencoders (VAEs) introduce a probabilistic framework that explicitly models latent variables and their distributions.
- We want a model that both reconstructs digits and can generate new digits by sampling.

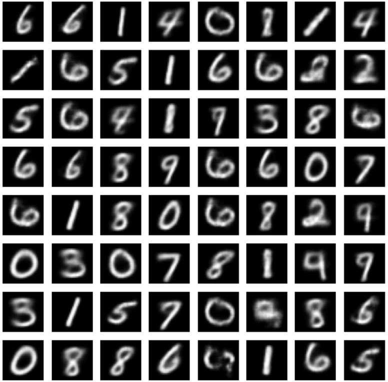
Dataset: MNIST

- MNIST handwritten digits
- 60,000 training / 10,000 test images
- Image space:

$$x \in [0, 1]^{28\times 28}$$

- Grayscale, normalized
- Input is normalized to $[0, 1]$, treated as Bernoulli likelihood for BCE.
- Unsupervised learning (labels unused)

VAE: Samples from N(0, I)



Problem Statement

- Given a dataset of handwritten digit images:

$$\mathcal{D} = \{x_i\}_{i=1}^{N}, \quad x_i \sim p_{\text{data}}(x), \quad x_i \in [0,1]^{28\times 28}$$

- Goal: learn a probabilistic latent-variable model that:
  - Encodes each image into a low-dimensional latent variable

$$z \in \mathbb{R}^d \quad (d \in \{2, 16\})$$

  - Accurately reconstructs the input image
  - Supports generation of new samples by sampling in latent space
- Assume a generative process:

$$z \sim p(z), \quad x \sim p_\theta(x \mid z)$$

- Objective:
  - Learn parameters θ such that

$$p_\theta(x) \approx p_{\text{data}}(x)$$

Challenges and Modeling Goals

- Direct maximization of the data likelihood

$$\log p_\theta(x)$$

  is intractable for latent-variable models
- The true posterior distribution

$$p(z \mid x)$$

  cannot be computed exactly
- Limitations of deterministic autoencoders:
  - Encode each input as a single latent point
  - No explicit probabilistic interpretation of the latent space
  - Random sampling in latent space is not meaningful
- Modeling goals of this project:
  - Learn a smooth and continuous latent space
  - Enable meaningful interpolation between images
  - Allow principled sampling by enforcing a known prior $p(z)$

Baseline Method: Autoencoder

- Baseline: deterministic convolutional autoencoder (AE)
- Architecture:

$$x \xrightarrow{\text{encoder } f_\phi} z \xrightarrow{\text{decoder } g_\theta} \hat{x}$$

- Encoder maps each input image to a single latent point:

$$z = f_\phi(x), \quad z \in \mathbb{R}^d$$

- Decoder reconstructs the input from the latent representation:

$$\hat{x} = g_\theta(z)$$

- Trained solely to minimize reconstruction error (no latent regularization)

Autoencoder Objective

- The autoencoder is trained by minimizing a reconstruction loss:

$$\mathcal{L}_{\text{AE}}(x) = \|x - \hat{x}\|$$

- In this work, we use binary cross-entropy (BCE):

$$\mathcal{L}_{\text{AE}}(x) = -\sum_i x_i \log \hat{x}_i + (1 - x_i) \log(1 - \hat{x}_i)$$

- BCE is computed **per pixel**, summed over the image, and averaged over the batch
- No probabilistic interpretation of the latent space
- No explicit regularization on z

Why Use an Autoencoder as Baseline?

- Provides a strong reference point for reconstruction quality
- Uses a similar encoder–decoder architecture and capacity
- Key differences compared to VAE:
  - Encodes inputs as deterministic latent points
  - Does not impose a prior distribution on the latent space
  - Random sampling in latent space does not yield meaningful outputs
- Highlights the fundamental trade-off:
  - **Autoencoder:** better reconstruction fidelity
  - **VAE:** structured latent space and generative capability

Why a Variational Autoencoder (VAE)?

- We need a model that supports **latent-variable modeling** and **data generation**.
- VAE assumes a latent generative process:

$$z \sim p(z), \quad x \sim p_\theta(x \mid z)$$

- Key advantages for this problem (MNIST images):
  - Learns a **continuous, structured latent space** for representation learning
  - Enables **sampling** by drawing $z \sim p(z)$ and decoding
  - Supports **interpolation** and qualitative analysis of learned representations
- Aligns with course focus: **latent-variable modeling** + **distribution modeling**.
- Regularization toward a Gaussian prior enables meaningful sampling, as demonstrated in the experiments

Why Not Other Generative Methods? (Course Alternatives)

- **GANs:** lack explicit likelihood and interpretable posterior $q(z|x)$
- **Normalizing Flows:** architectural constraints due to invertibility
- **Diffusion Models:** computationally expensive for insight-focused analysis
- **Neural ODE / CNFs:** designed for continuous-time dynamics, not required here

VAE Objective: Evidence Lower Bound (ELBO)

- The true posterior $p_\theta(z \mid x)$ is intractable
- Introduce a variational approximation:

$$q_\phi(z \mid x) = \mathcal{N}\big(\mu_\phi(x), \mathrm{diag}(\sigma^2_\phi(x))\big)$$

- Optimize the Evidence Lower Bound (ELBO):

$$\log p_\theta(x) \geq \underbrace{\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x \mid z)]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q_\phi(z \mid x) \,\|\, p(z))}_{\text{regularization to prior}}$$

- In practice, minimize the negative ELBO:

$$\mathcal{L}_{\mathsf{VAE}} = \mathcal{L}_{\mathsf{recon}} + \beta\, \mathcal{L}_{\mathsf{KL}}, \quad \beta = 1$$

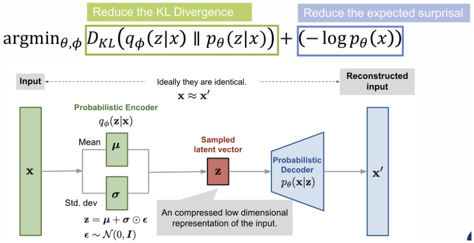
- Prior distribution: $p(z) = \mathcal{N}(0, I)$

VAE Architecture (Encoder–Decoder)

- **Encoder** $q_\phi(z \mid x)$
  - Convolutional layers for feature extraction
  - Dense layers output $\mu_\phi(x)$ and $\log \sigma^2_\phi(x)$
- **Reparameterization trick**

$$z = \mu + \sigma \odot \epsilon, \quad \epsilon \sim \mathcal{N}(0, I)$$

- **Decoder** $p_\theta(x \mid z)$
  - Dense + reshape + ConvTranspose layers
  - Sigmoid output models pixels in $[0, 1]$

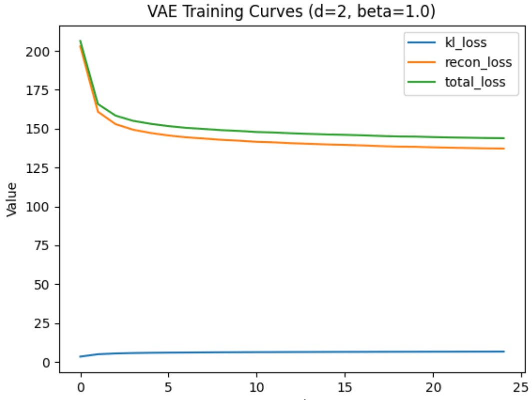


Training Procedure & Reproducibility

- **Data preprocessing:**
  - MNIST images normalized to $[0, 1]$ and reshaped to $(28, 28, 1)$
  - Labels not used during training (unsupervised learning)
- **Optimization:**
  - Optimizer: Adam, learning rate 1×10^{-3}
  - Batch size: 128, epochs: 25
  - Reconstruction loss: binary cross-entropy (BCE) over pixels
  - Regularization: $D_{\mathrm{KL}}(q_\phi(z \mid x) \parallel p(z))$
- **Experimental settings:**
  - VAE with latent dimensions $d = 2$ and $d = 16$
  - Deterministic autoencoder baseline with matched capacity
- **Reproducibility:**
  - Fixed random seed
  - All hyperparameters and evaluation metrics recorded

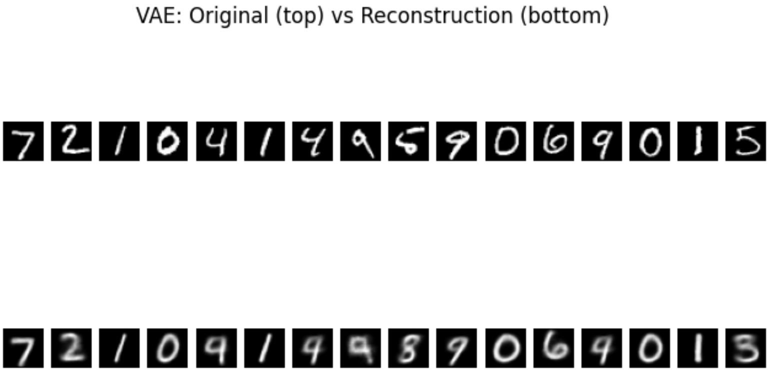
Training Dynamics (VAE, d=2)

- Model trained with latent dimension $d = 2$ and $\beta = 1.0$
- Total loss decreases smoothly and stabilizes
- Reconstruction loss dominates early training
- KL divergence increases initially, then stabilizes



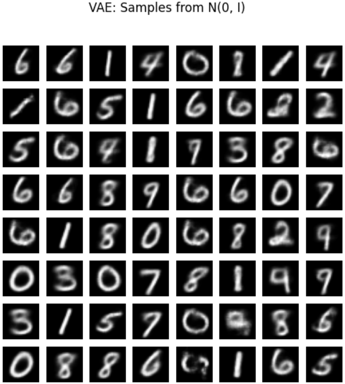
Reconstruction Quality (VAE, d=2)

- Top row: original MNIST images
- Bottom row: VAE reconstructions
- Digits remain recognizable, but fine details are smoothed
- Blurriness is expected due to KL regularization



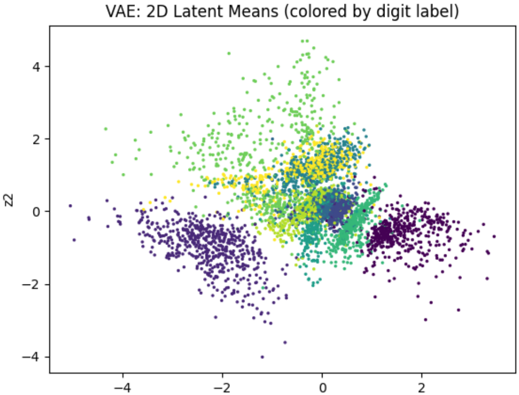
Generative Sampling from the Prior

- Samples generated by drawing:
$$z \sim \mathcal{N}(0, I)$$
- Decoder produces diverse and recognizable digits
- Some ambiguity reflects limited capacity of $d = 2$



Latent Space Structure (d=2)

- Each point represents the latent mean $\mu(x)$
- Colored by digit label (labels not used during training)
- Digits form partially separated clusters
- Overlaps reflect visual similarity and unsupervised learning

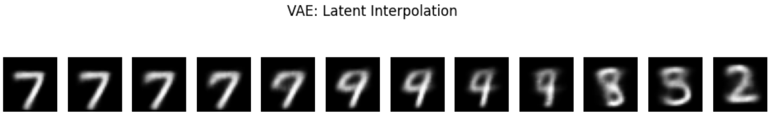


Latent Space Interpolation

- Interpolation between two test images in latent space
- Linear interpolation between latent codes:

$$z_\alpha = (1 - \alpha)z_1 + \alpha z_2$$

- Produces smooth semantic transitions



Analysis of VAE Behavior (d = 2)

- The VAE successfully balances reconstruction accuracy and latent regularization:
  - Reconstruction loss decreases steadily during training
  - KL divergence stabilizes at a non-zero value
- This indicates that the latent variables are actively used rather than ignored
- The learned 2D latent space is:
  - Continuous and smooth
  - Partially clustered by digit identity
  - Not explicitly supervised, yet semantically meaningful
- Smooth latent interpolations confirm that nearby latent points correspond to visually similar digits

Limitations and Trade-offs Observed

- Strong compression with $d = 2$ limits reconstruction fidelity
  - Fine-grained digit details are smoothed
  - Some generated samples are ambiguous
- This reflects the fundamental VAE trade-off:

$$\text{Reconstruction quality} \quad \leftrightarrow \quad \text{Latent structure}$$

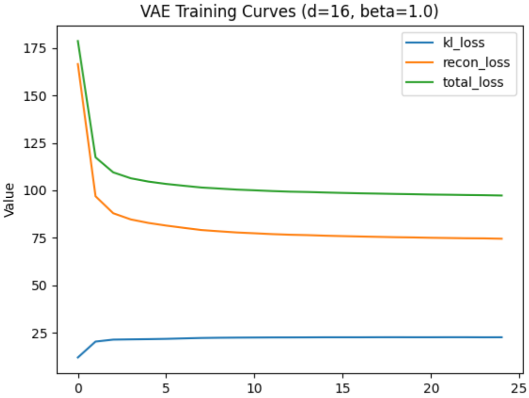
- Compared to the deterministic autoencoder:
  - AE achieves sharper reconstructions
  - VAE provides a structured latent space suitable for sampling
- Motivates exploring higher latent dimensions to increase model capacity

Why Study Latent Dimension?

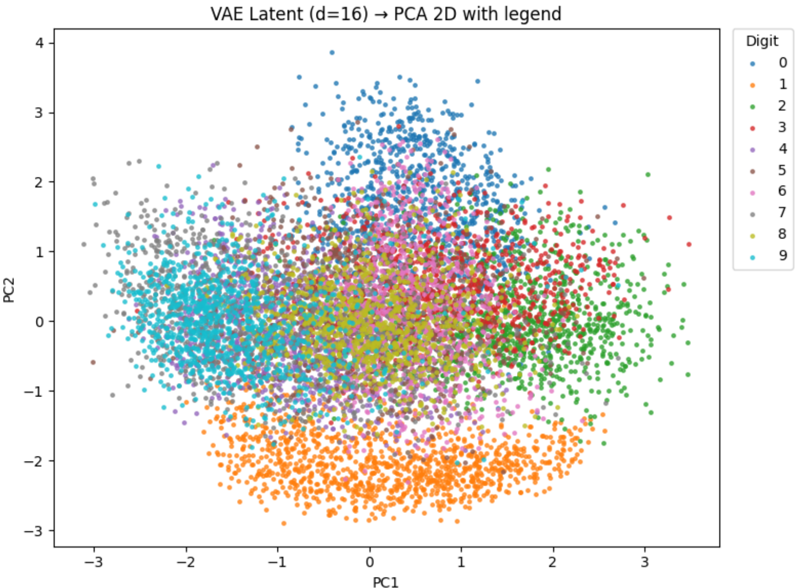
- Latent dimension d controls the information capacity of the model
- Small d:
  - Strong compression
  - Encourages structured, interpretable latent spaces
- Larger d:
  - Higher reconstruction capacity
  - Potentially weaker regularization effect
- Goal: understand the trade-off between reconstruction quality and latent structure

Training Dynamics: d = 2 vs d = 16

- Both models converge smoothly and stably
- Increasing latent dimension significantly reduces reconstruction loss
- KL divergence increases with higher d, reflecting increased latent usage
- Total ELBO loss stabilizes for both settings



Latent Space Visualization (d = 16, PCA Projection)



Latent Space Visualization (d = 16, PCA Projection)

- Latent representations extracted from the VAE encoder with $d = 16$
- High-dimensional latent means projected to 2D using Principal Component Analysis (PCA)
- Points are colored by digit label (labels not used during training)
- Observations:
  - Different digits form partially separated regions
  - Significant overlap reflects unsupervised learning and shared visual features
  - Structure indicates that semantic information is captured in the latent space
- PCA is used only for visualization and does not affect training

Quantitative Comparison: d = 2 vs d = 16

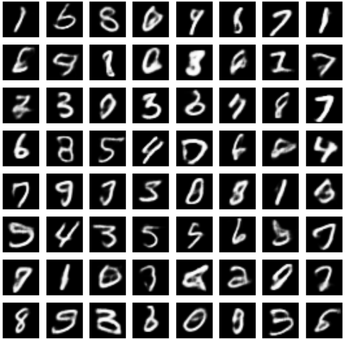
Latent Dim	Recon Loss	KL Divergence	Total Loss
$d = 2$	$\sim$137.0	$\sim$6.0	$\sim$143.0
$d = 16$	74.99	22.52	97.51

- Larger latent dimension improves reconstruction accuracy
- KL divergence increases, indicating more active latent variables
- Confirms capacity–regularization trade-off

Qualitative Comparison: Generation and Interpolation

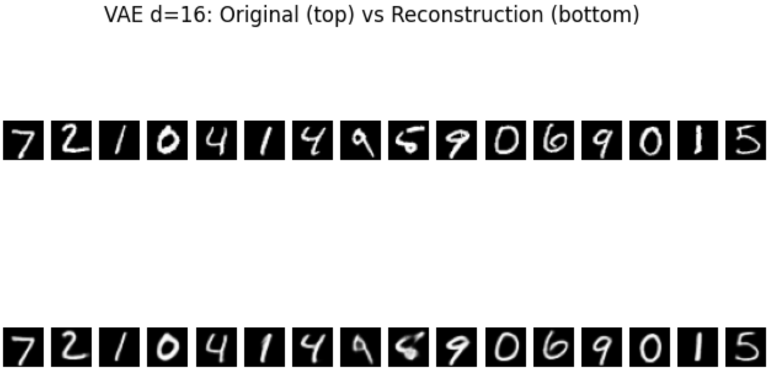
- $d = 16$ produces sharper and more detailed samples
- Interpolations remain smooth and semantically meaningful
- Higher capacity reduces ambiguity in generated digits

VAE d=16: Samples from N(0, I)



Reconstruction Quality (VAE, d = 16)

- Top row: original MNIST images
- Bottom row: VAE reconstructions with latent dimension $d = 16$
- Reconstructions are significantly sharper than for $d = 2$
- Higher latent capacity allows preservation of fine-grained details



Analysis of Trade-offs

- $d = 2$:
  - Strong regularization
  - Highly interpretable latent space
  - Lower reconstruction fidelity
- $d = 16$:
  - Higher reconstruction quality
  - More expressive latent representation
  - Slightly reduced interpretability
- Choice of d depends on task objectives:
  - Visualization vs generation fidelity

Conclusion & Takeaways

- Variational Autoencoders provide a principled framework for latent-variable modeling
- KL regularization enables smooth, structured latent spaces
- Compared to autoencoders:
  - AE excels at reconstruction
  - VAE enables meaningful sampling and interpolation
- Latent dimension controls the trade-off between structure and fidelity
- Experiments validate theoretical properties of VAEs in practice